Classic Kriging versus Kriging with Bootstrapping or Conditional Simulation

Mehdad, E.; Kleijnen, Jack P.C.

Publication date:
2014

Document Version
Early version, also known as pre-print

CentER

Discussion Paper

No. 2014-076

CLASSIC KRIGING VERSUS KRIGING WITH BOOTSTRAPPING OR CONDITIONAL SIMULATION: CLASSIC KRIGING'S ROBUST CONFIDENCE INTERVALS AND OPTIMIZATION

By

Ehsan Mehdad, Jack P.C. Kleijnen

2 December, 2014

This is a revised version of CentER Discussion Paper

No. 2013-038

ISSN 0924-7815
ISSN 2213-9532

Classic Kriging versus Kriging with bootstrapping or conditional simulation: classic Kriging's robust confidence intervals and optimization

Ehsan Mehdad
Jack P.C. Kleijnen
Tilburg School of Economics and Management, Tilburg University
Postbox 90153, 5000 LE Tilburg, The Netherlands

November 24, 2014

Abstract

Kriging is a popular method for estimating the global optimum of a simulated system. Kriging approximates the input/output function of the simulation model. Kriging also estimates the variances of the predictions of outputs for input combinations not yet simulated. These predictions and their variances are used by "efficient global optimization"(EGO), to balance local and global search. This article focuses on two related questions: (1) How to select the next combination to be simulated when searching for the global optimum? (2) How to derive confidence intervals for outputs of input combinations not yet simulated? Classic Kriging simply plugs the estimated Kriging parameters into the formula for the predictor variance, so theoretically this variance is biased. This article concludes that practitioners may ignore this bias, because classic Kriging gives acceptable confidence intervals and estimates of the optimal input combination. This conclusion is based on bootstrapping and conditional simulation.

Keywords: Simulation, Optimization, Kriging, Bootstrap, Conditional simulation

JEL: C0, C1, C9, C15, C44

1 Introduction

In this article we address the following two related questions that arise in simulation, especially when the simulation is "computationally expensive":

1. How to derive a *confidence interval* (CI) for the output of a "new" combination of simulation inputs that is not yet simulated?

2. How to select the next combination that is to be simulated, when searching for the *optimal* combination?

Question 1 (Q1) arises in sensitivity analysis or "what if" analysis. Question 2 (Q2) arises in "simulation optimization", which aims at finding the input combination—also called scenario or point—that gives the minimal simulation output (response); we limit our optimization to unconstrained problems, like many authors do.

To answer these two questions, simulation analysts often use metamodels, also called approximations, emulators, surrogates, etc. A popular type of metamodel is a *Kriging* or *Gaussian process* (GP) model; also see the survey in Kleijnen (2009). *Classic Kriging* (CK)—as we call it—estimates the variance of its predictor by plugging-in the estimated parameters (say) $\widehat{\boldsymbol{\psi}}$ of the assumed stationary GP (we assume a GP with parameter vector $\boldsymbol{\psi}$ consisting of the constant mean β_0, the constant variance τ^2, and the correlation vector $\boldsymbol{\theta}$; see Section 2). Obviously, plugging-in $\widehat{\boldsymbol{\psi}}$ makes the Kriging predictor nonlinear so $s^2(\mathbf{x})$, the classic variance estimator of the Kriging predictor at point $\mathbf{x}$, is *biased*. Indeed, (Jones et al., 1998, p. 463) states: "This theoretical sleight of hand appears to have no serious consequences, although it probably leads to a slight underestimation of prediction error in small samples". To the best of our knowledge, the literature has not tested this conjecture. We therefore (empirically) compare the CI of CK and alternative CIs (see Q1). Note that Goel et al. (2006) also comment on the classic variance estimator, and propose cross-validation (whereas we propose BK or CS).

Moreover, $s^2(\mathbf{x})$ is also used in *efficient global optimization* (EGO), which is a well-known sequential method that balances local and global search; i.e., EGO combines exploitation and exploration. The classic EGO article is Jones et al. (1998); recent articles are Picheny et al. (2013); Viana et al. (2013).

To answer Q1 and Q2, we apply *parametric bootstrapping* in this article. Bootstrapping in general—including both parametric and nonparametric or distribution-free bootstrapping—is discussed in Efron and Tibshirani (1993); additional recent references are given in (Kleijnen, 2008, pp.81, 84). We compare the following alternative methods with each other and with CK:

- *bootstrapped Kriging* (BK), originally proposed in Den Hertog et al. (2006) to examine $s^2(\mathbf{x})$ and in Kleijnen et al. (2012) to examine EGO;
- *conditional simulation* (CS), which is popular in the French literature on Kriging; see the references in (Wackernagel, 2003, p. 188) and Section 2.3.

A preliminary version of our research was presented in Kleijnen and Mehdad (2013), comparing the estimated variances of the Kriging predictors in CK, BK, and CS and their effects on EGO. Now we investigate the role of this variance in the CI (see again Q1); i.e., what are the coverages and lengths of the CIs when using these three methods? Moreover, for these CIs we use either the classic *parametric* method assuming the predictor is unbiased and normally distributed—even though the predictor is nonlinear—and a *distribution-free* method using the percentile method; the percentile method was originally discussed by (Efron and Tibshirani, 1993, p. 168-177) for bootstrapping in general. Furthermore, we present details on additional examples; i.e., to the detailed one-dimensional example in Kleijnen and Mehdad (2013) we add three well-known higher-dimensional examples.

We limit our research to *deterministic* simulation, which is popular in engineering, and will be the basis for our future research on random (stochastic) discrete-event simulation. Our main conclusion will be that CK seems quite robust; i.e., (i) BK and CS give CIs with coverages and lengths that are not significantly better than CK; (ii) EGO with BK or CS may or may not give a bootstrap sample that performs better in expensive simulation with small samples.

Besides this introductory section, our article comprises the following sections. In Section 2 we summarize CK (including our terminology and symbols), BK, and CS; we also study the effects of dimensionality on the predictor variance. In Section 3 we present CIs for Kriging predictors. In Section 4 we first summarize EGO based on CK, BK, and CS, and we include a new EGO variant that uses CS with a distribution-free method; next we give numerical examples. In Section 5 we present our conclusions and topics for further research.

2 CK, BK, and CS

In this section we summarize CK, BK, and CS—based on Kleijnen and Mehdad (2013). We add a proof for the asymptotic behavior of CS and BK, and experimental results on the effects of dimensionality on the predictor variance.

2.1 Classic Kriging

To estimate a Kriging metamodel of an underlying simulation model, we simulate (say) k points $\mathbf{x}_i$ $(i = 1, \ldots, k)$, which combine $d \geq 1$ simulation inputs. This simulation gives the output w_i. Hence, the set of input/output (I/O) data is $(\mathbf{X}, \mathbf{w})$ where $\mathbf{X}$ denotes the $k \times d$ matrix with rows $\mathbf{x}_i$, and $\mathbf{w} = (w_1, \ldots, w_k)^\top$. A rule-of-thumb for k is that a valid Kriging metamodel requires $k = 10d$ points when these points are selected through Latin hypercube sampling (LHS); see Loeppky et al. (2009).

In deterministic simulation, Kriging is an *exact interpolator*; i.e., the Kriging predictions $y(\mathbf{x}_i) = y_i$ equal the corresponding observed simulation outputs $w(\mathbf{x}_i) = w_i$ for the k "old" input combinations $\mathbf{x}_i$.

Ordinary Kriging assumes that its output $y(\mathbf{x})$ is a realization of the random process

$$Y(\mathbf{x}) = \beta_0 + M(\mathbf{x}) \tag{1}$$

with the constant mean β_0 (also denoted by μ) and the stochastic process $M(\mathbf{x})$ with covariance matrix $\mathbf{\Sigma}_M$, where the covariance between $M(\mathbf{x})$ and $M(\mathbf{x}')$ is $\Sigma_M(\mathbf{x}, \mathbf{x}') = \tau^2 R_M(\mathbf{x}, \mathbf{x}')$ with constant process variance τ^2 and correlation matrix $\mathbf{R}_M$; more precisely, $E[M(\mathbf{x})] = 0$ and the correlation between $\mathbf{x}$ and $\mathbf{x}'$

depends only on the distance $|\mathbf{x} - \mathbf{x}'|$. In this article we use the most popular R_M in simulation (also see Xie et al. (2010)); namely, the *Gaussian* correlation function in product form:

$$R_M(\mathbf{x}, \mathbf{x}', \boldsymbol{\theta}) = \prod_{j=1}^{d} \exp[-\theta_j (x_j - x'_j)^2] \text{ with } \theta_j > 0. \tag{2}$$

To select $\widehat{Y}(\mathbf{x}_0)$—the *predictor* of the output at a new point $\mathbf{x}_0$—the criterion is the mean squared prediction error (MSPE):

$$\text{MSPE}[\widehat{Y}(\mathbf{x}_0)] = E[\widehat{Y}(\mathbf{x}_0) - w(\mathbf{x}_0)]^2. \tag{3}$$

The minimum of (3) is determined by the following $(1+k)$-dimensional Gaussian or Normal distribution:

$$\begin{pmatrix} Y(\mathbf{x}_0) \\ Y(\mathbf{x}) \end{pmatrix} \sim \mathrm{N}_{1+k}\left[\beta_0 \mathbf{1}_{1+k}, \begin{pmatrix} \tau^2 & \boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot)^\top \\ \boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot) & \boldsymbol{\Sigma}_M \end{pmatrix}\right] \tag{4}$$

where $\mathbf{1}_{1+k}$ denotes the vector with all its $(1 + k)$ elements equal to 1, and $\boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot)$ denotes the vector with the covariances between the output of the "new" point $\mathbf{x}_0$ and the outputs of the k old points $\mathbf{x}_i$ so its elements are $\text{Cov}[M(\mathbf{x}_0), M(\mathbf{x}_i)]$. The predictor $\widehat{Y}(\mathbf{x}_0)$ is required to be linear (say) $\widehat{Y}(\mathbf{x}_0) = \mathbf{a}^\top Y(\mathbf{x})$ and unbiased so $E[\widehat{Y}(\mathbf{x}_0)|Y(\mathbf{x})] = E[Y(\mathbf{x}_0)|Y(\mathbf{x})]$. The *best linear unbiased predictor* (BLUP) can be derived to be

$$\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi}) = \beta_0 + \boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot)^\top \boldsymbol{\Sigma}_M^{-1}[Y(\mathbf{x}) - \beta_0 \mathbf{1}_k] \tag{5}$$

where we introduce the symbol $\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi})$ to emphasize that the predictor depends on $\boldsymbol{\psi} = (\beta_0, \tau^2, \boldsymbol{\theta}^\top)^\top$, which denotes the vector with all the GP parameters. Together, (3) and (5) give the $\text{MSPE}[\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi})]$. Because $\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi})$ is unbiased, this $\text{MSPE}[\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi})]$ equals the predictor variance $\sigma^2[\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi})]$. It can be derived that

$$\sigma^2[\widehat{Y}(\mathbf{x}_0, \boldsymbol{\psi})] = \tau^2 - \boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot)^\top \boldsymbol{\Sigma}_M^{-1} \boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot) + \frac{[1 - \mathbf{1}_k^\top \boldsymbol{\Sigma}_M^{-1} \boldsymbol{\Sigma}_M(\mathbf{x}_0, \cdot)]^2}{\mathbf{1}_k^\top \boldsymbol{\Sigma}_M^{-1} \mathbf{1}_k}. \tag{6}$$

In practice, however, $\boldsymbol{\psi}$ is unknown and is *estimated.* CK uses the maximum likelihood estimators (MLEs) $\widehat{\boldsymbol{\psi}} = (\widehat{\beta}_0, \widehat{\tau}^2, \widehat{\boldsymbol{\theta}}^\top)^\top$. These MLEs follow from the log-likelihood function, which follows from the distribution (4). This function is rather complicated, so Kriging computes these MLEs numerically through a constrained maximization algorithm. Different Kriging packages use different algorithms. We use the popular free MATLAB Kriging toolbox *DACE*—developed by Lophaven et al. (2002)—which applies the Hooke-Jeeves algorithm.

The predictor for $\mathbf{x}_0$ with plugged-in $\widehat{\boldsymbol{\psi}}$ follows from (5):

$$\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) = \widehat{\beta}_0 + \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1}[Y(\mathbf{x}) - \widehat{\beta}_0 \mathbf{1}_k]. \tag{7}$$

Because this predictor is *nonlinear*, its MSPE and variance are unknown. We define $\widehat{\sigma}^2_{\text{CK}}[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})]$ that follows from (6):

$$\widehat{\sigma}^2_{\text{CK}}[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})] = \widehat{\tau}^2 - \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1} \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot) + \frac{[1 - \mathbf{1}_k^\top \widehat{\boldsymbol{\Sigma}}_M^{-1} \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot)]^2}{\mathbf{1}_k^\top \widehat{\boldsymbol{\Sigma}}_M^{-1} \mathbf{1}_k}. \quad (8)$$

We denote this $\widehat{\sigma}^2_{\text{CK}}[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})]$ by $\widehat{\sigma}^2_{\text{CK}}$, and observe that $\widehat{\sigma}^2_{\text{CK}}$ was denoted by $s^2(\mathbf{x})$ in the quotes in Section 1. We conjecture that $\widehat{\sigma}^2_{\text{CK}}$ underestimates the true variance σ^2_{CK}, because it ignores the randomness of the MLEs, but we do not know how serious this bias is. We therefore derive alternative estimators in the next two sections.

2.2 Bootstrapped Kriging

BK was developed by Den Hertog et al. (2006) to estimate the predictor variance as a function of $\mathbf{x}_0$. It is well-known that as $\mathbf{x}_0$ gets closer to an old point $\mathbf{x}_i$, its predictor variance decreases and becomes zero when the new point and an old point coincide (Kriging is an exact interpolator). Furthermore, N_{1+k} in (4) implies that the distribution of the new output given the k old outputs is a conditional normal distribution.

The bootstrap literature denotes bootstrapped data by the superscript $*$ (e.g., $\mathbf{w}^*$). Bootstrapped estimators (e.g. $\widehat{\boldsymbol{\psi}}^*$) are defined analogously to the definitions of the original estimators (e.g., $\widehat{\boldsymbol{\psi}}$), but the bootstrapped estimators are computed from the bootstrapped data (e.g., $\mathbf{w}^*$) instead of the original data (e.g. $\mathbf{w}$). The bootstrap sample size is denoted by B (the standard symbol in the bootstrap literature). The b^{th} bootstrap observation in the bootstrap sample is denoted by the subscript b with $b = 1, \ldots, B$.

The BK algorithm has the following steps.

1. Use $\text{N}_k\left(\widehat{\beta_0}\mathbf{1}_k, \widehat{\boldsymbol{\Sigma}}_M\right)$ B times to sample the k old points $\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}}) = (w^*_{1;b}(\mathbf{X}, \widehat{\boldsymbol{\psi}}), \ldots, w^*_{k;b}(\mathbf{X}, \widehat{\boldsymbol{\psi}}))^\top$ where $\widehat{\boldsymbol{\psi}}$ is estimated from the old I/O data $(\mathbf{X}, \mathbf{w})$. For each new point $\mathbf{x}_0$ repeat steps 2 through 4 B times.

2. Given the k old points $\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}})$ of step 1, sample the new point $w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$ from the following conditional normal distribution:

$$\text{N}\left[\widehat{\beta}_0 + \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1}[Y(\mathbf{x}) - \widehat{\beta}_0\mathbf{1}_k], \widehat{\tau}^2 - \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1} \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot)\right]. \quad (9)$$

3. Using the k old bootstrapped points $\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}})$ of step 1, compute the bootstrapped MLE $\widehat{\boldsymbol{\psi}}_b^*$. Next calculate the bootstrapped predictor

$$\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*) = \widehat{\beta}^*_{0;b} + \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot, \widehat{\boldsymbol{\psi}}_b^*)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1}(\widehat{\boldsymbol{\psi}}_b^*)[\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}}) - \widehat{\beta}^*_{0;b}\mathbf{1}_k].$$

4. Given $\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)$ of step 3 and $w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$ of step 2, compute the bootstrap estimator of the squared prediction error (SPE):

$$\mathrm{SPE}_b = \mathrm{SPE}[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)] = [\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*) - w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})]^2.$$

5. Given the B bootstrap estimators SPE_b ($b = 1, \ldots, B$) resulting from steps 1 through 4, compute the bootstrap estimator of $\mathrm{MSPE}[\widehat{Y}(\mathbf{x}_0)]$ (this MSPE was defined in (3):

$$\mathrm{MSPE}[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}^*)] = \frac{\sum_{b=1}^{B} \mathrm{SPE}_b}{B}. \tag{10}$$

Ignoring the bias of the BK predictor $\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}^*)$, (10) gives $\widehat{\sigma}^2[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}^*)]$ which is the bootstrap estimator of $\sigma^2[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})]$. We abbreviate $\widehat{\sigma}^2[\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}^*)]$ to $\widehat{\sigma}^2_{\mathrm{BK}}$.

Obviously, the standard error (SE) of our estimator $\widehat{\sigma}^2_{\mathrm{BK}}$ is

$$\mathrm{SE}(\widehat{\sigma}^2_{\mathrm{BK}}) = \sqrt{\frac{\sum_{b=1}^{B}(\mathrm{SPE}_b - \mathrm{MSPE})^2}{(B-1)B}}.$$

Applying t_{B-1} (t-statistic with $B-1$ degrees of freedom) gives the following two-sided symmetric $(1-\alpha)$ CI:

$$P\{\sigma^2_{\mathrm{CK}} \in \widehat{\sigma}^2_{\mathrm{BK}} \pm t_{B-1;\alpha/2}\mathrm{SE}(\widehat{\sigma}^2_{\mathrm{BK}})\} = 1 - \alpha. \tag{11}$$

If $B \uparrow \infty$, then $t_{B-1;\alpha/2} \downarrow z_{\alpha/2}$ where $z_{\alpha/2}$ denotes the $\alpha/2$ quantile of the standard normal variable $z \sim \mathrm{N}(0, 1)$.

(Kleijnen and Mehdad, 2013, Figure 1) illustrates BK for (Forrester et al., 2008, p. 83)'s test function defined in (17). This illustration shows that each of the B samples has its own old output values, and that $B = 20{,}000$ seems to confirm the conjecture $\widehat{\sigma}^2_{\mathrm{BK}} \gg \widehat{\sigma}^2_{\mathrm{CK}}$. (Yin et al. (2010) also find empirically that their Bayesian approach accounting for the randomness of the estimated Kriging parameters gives a wider CI—and hence higher coverage—than an approach that ignores this estimation.)

2.3 Conditional simulation

We adapt the following CS algorithm from Kleijnen and Mehdad (2013), copying steps 1 through 3 of the BK algorithm in the preceding section.

1. Use $\mathrm{N}_k\left(\widehat{\beta_0}\mathbf{1}_k, \widehat{\boldsymbol{\Sigma}}_M\right)$ B times to sample the k old points $\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}}) = (w_{1;b}^*(\mathbf{X}, \widehat{\boldsymbol{\psi}}), \ldots, w_{k;b}^*(\mathbf{X}, \widehat{\boldsymbol{\psi}}))^\top$ where $\widehat{\boldsymbol{\psi}}$ is estimated from the old I/O data $(\mathbf{X}, \mathbf{w})$. For each new point $\mathbf{x}_0$ repeat steps 2 through 4 B times.

2. Given the k old points $\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}})$ of step 1, sample the new point $w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$ from the conditional normal distribution

$$\mathrm{N}\left[\widehat{\beta}_0 + \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0,\cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1}[Y(\mathbf{x}) - \widehat{\beta}_0\mathbf{1}_k], \widehat{\tau}^2 - \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0,\cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1} \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0,\cdot)\right],$$

which equals (9).

3. Using the k old bootstrapped points $\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}})$ of step 1, compute the bootstrapped MLE $\widehat{\boldsymbol{\psi}}_b^*$. Next calculate the bootstrapped predictor

$$\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*) = \widehat{\beta}_{0;b}^* + \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0, \cdot, \widehat{\boldsymbol{\psi}}_b^*)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1}(\widehat{\boldsymbol{\psi}}_b^*)[\mathbf{w}_b^*(\mathbf{X}, \widehat{\boldsymbol{\psi}}) - \widehat{\beta}_{0;b}^*\mathbf{1}_k]. \quad (12)$$

4. Combining the CK estimator (7) and the BK estimator (12), compute the CS output at the new point:

$$\widehat{Y}_{\mathrm{CS}}(\mathbf{x}_0, b) = \widehat{\beta}_0 + \widehat{\boldsymbol{\Sigma}}_M(\mathbf{x}_0,\cdot)^\top \widehat{\boldsymbol{\Sigma}}_M^{-1}(\mathbf{w} - \widehat{\beta}_0\mathbf{1}_k) + [w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) - \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)]. \quad (13)$$

5. Given the B CS estimators $\widehat{Y}_{\mathrm{CS}}(\mathbf{x}_0, b)$ $(b = 1, \ldots, B)$ defined in (13), which result from steps 1 through 4, compute the CS estimator of MSPE$[\widehat{Y}(\mathbf{x}_0)]$:

$$\widehat{\sigma}^2[\widehat{Y}_{\mathrm{CS}}(\mathbf{x}_0)] = \frac{\sum_{b=1}^B [\widehat{Y}_{\mathrm{CS}}(\mathbf{x}_0, b) - \overline{\widehat{Y}}_{\mathrm{CS}}(\mathbf{x}_0)]^2}{B-1} \text{ with} \quad (14)$$
$$\overline{\widehat{Y}}_{\mathrm{CS}}(\mathbf{x}_0) = \frac{\sum_{b=1}^B \widehat{Y}_{\mathrm{CS}}(\mathbf{x}_0, b)}{B}.$$

We abbreviate $\widehat{\sigma}^2[\widehat{Y}_{\mathrm{CS}}(\mathbf{x}_0)]$ to $\widehat{\sigma}^2_{\mathrm{CS}}$. Now we prove that $\widehat{\sigma}^2_{\mathrm{CS}} \le \widehat{\sigma}^2_{\mathrm{BK}}$. We ignore the first two terms in the right-hand side of (13) because these terms do not depend on b. So we obtain

$$\begin{aligned}
&\widehat{\sigma}^2_{\mathrm{CS}} = \widehat{\sigma}^2[w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) - \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)] = \\
&\quad \frac{\sum_{b=1}^B [w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) - \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)]^2}{B} - \left(\frac{\sum_{b=1}^B [w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) - \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)]}{B}\right)^2 = \\
&\quad \widehat{\sigma}^2_{\mathrm{BK}} - \left(\frac{\sum_{b=1}^B [w_b^*(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) - \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}_b^*)]}{B}\right)^2. \qquad (15)
\end{aligned}$$

Because the second term in the last equality is a square, this term is non-negative so $\widehat{\sigma}^2_{\mathrm{CS}} \le \widehat{\sigma}^2_{\mathrm{BK}}$. Actually, this term is the bootstrapped estimator of the mean prediction error (MPE). In practice, we only know that this MPE is not exactly zero (because the Kriging metamodel is not perfect), so $\widehat{\sigma}^2_{\mathrm{CS}} < \widehat{\sigma}^2_{\mathrm{BK}}$, but we do not know how much smaller $\widehat{\sigma}^2_{\mathrm{CS}}$ is than $\widehat{\sigma}^2_{\mathrm{BK}}$. We therefore derive a two-sided asymmetric $(1-\alpha)$ CI for σ^2_{CK}, using $\widehat{\sigma}^2_{\mathrm{CS}}$ and the chi-square statistic χ^2_{B-1} (this CI replaces (11) for BK):

$$P\left\{\frac{(B-1)\widehat{\sigma}^2_{\mathrm{CS}}}{\chi^2_{B-1;1-\alpha/2}} \le \sigma^2_{\mathrm{CK}} \le \frac{(B-1)\widehat{\sigma}^2_{\mathrm{CS}}}{\chi^2_{B-1;\alpha/2}}\right\} = 1 - \alpha. \quad (16)$$

CS for Forrester's function is illustrated in (Kleijnen and Mehdad, 2013, Figure 3). That plot suggests that $\widehat{\sigma}^2_{\text{CS}}$ indeed tends to exceed $\widehat{\sigma}^2_{\text{CK}}$; results for $B = 20{,}000$ seem to confirm this conjecture. Furthermore, this plot suggests that $\widehat{\sigma}^2_{\text{BK}}$ indeed tends to exceed $\widehat{\sigma}^2_{\text{CS}}$; for $B \uparrow \infty$ the two estimators tend to the same asymptotic value (see (15)); for small bootstrap samples, CS does not give a significantly smaller value. These results seem reasonable, because both CS and BK use $\widehat{\boldsymbol{\psi}}$, which is the *sufficient* statistic of the GP computed from the same $(\mathbf{X}, \mathbf{w})$. Computationally, CS and BK have the same requirements, because CS includes bootstrapping similar to BK. Conceptually, we prefer CS because CS implies that at the old points its predictors equal the observed (simulated) outputs: $\widehat{Y}_{\text{CS}}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}^*_b) = w(\mathbf{x}_0)$. Moreover, we shall see that the CS point predictors $\widehat{Y}_{\text{CS}}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}^*_b)$—besides the predictor variance σ^2_{CS}—may be used in EGO.

2.4 Dimensionality effects

We conjecture that as the dimensionality d of the Kriging model increases, the bias of CK increases. To measure this bias, we consider a scale-free measure; namely, the coefficient of variation (CV) σ/μ. For the predictor variance, this CV becomes $\nu = \sigma[\widehat{y}(\mathbf{x})]/E[\widehat{y}(\mathbf{x})]$. Because Kriging assumes that the predictor has no bias, we use $E[\widehat{y}(\mathbf{x})] = y(\mathbf{x})$. For some $\mathbf{x}$, however, our test functions may have $y(\mathbf{x}) < 0$ or $y(\mathbf{x}) \approx 0$ (divide by zero). We therefore move the I/O plot "upwards"; i.e., we replace $y(\mathbf{x})$ by $y(\mathbf{x}) + c$ with $c = 1.1\,|y_{\text{opt}}|$ where y_{opt} denotes the true minimum output (which we know for the test function) and our choice of the factor 1.1 is rather arbitrary. Altogether we compute the *estimated modified CV* $\widehat{\nu} = \widehat{\sigma}[\widehat{y}(\mathbf{x})]/\{y(\mathbf{x}) + c\}$ so $\widehat{\nu} > 0$.

We compute this $\widehat{\nu}$ for both CK and CS, for the following four popular *test functions* with different d (which we shall also use in the section on EGO; these functions—except for the first one—are also discussed in `http://www-optima.amp.i.kyoto-u.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page364.htm` and `http://www.sfu.ca/~ssurjano/index.html`).

The *Forrester* function with $d = 1$:

$$w(x) = (6x - 2)^2 \sin(12x - 4) \tag{17}$$

with $0 \le x \le 1$, one local minimum at $x = 0.01$, and one global minimum at $x_{\text{opt}} = 0.7572$ with output $w = -6.02074$.

The *camel-back* function with $d = 2$:

$$w(x_1, x_2) = 4x_1^2 - 2.1x_1^4 + x_1^6/3 + x_1x_2 - 4x_2^2 + 4x_2^4 \tag{18}$$

with $-2 \le x_1 \le 2$ and $-1 \le x_2 \le 1$, two global minima $(\pm 0.089842, \mp 0.712656)^\top$ with $w = -1.031628$, and two additional local minima; for details see (Törn and Žilinskas, 2008, pp. 183-184).

The *Hartmann-3* function with $d = 3$:

$$w(x_1, x_2, x_3) = -\sum_{i=1}^{4} \alpha_i \exp[-\sum_{j=1}^{3} A_{ij}(x_j - P_{ij})^2] \tag{19}$$

with $0 \leq x_i \leq 1$ $(i = 1, 2, 3)$; parameters $\alpha = (1.0, 1.2, 3.0, 3.2)^\top$ and A_{ij} and P_{ij} given in Table 1; a global minimum at $(0.114614, 0.555649, 0.852547)^\top$ with $w = -3.86278$, and three additional local minima.

Table 1: Parameters A_{ij} and P_{ij} of the Hartmann-3 function

A_{ij}			P_{ij}		
3	10	30	0.36890	0.1170	0.26730
0.1	10	35	0.46990	0.43870	0.74700
3	10	30	0.10910	0.87320	0.55470
0.1	10	35	0.03815	0.57430	0.88280

The*Hartmann-6* function with $d = 6$:

$$w(x_1, \dots, x_6) = -\sum_{i=1}^{4} c_i \exp[-\sum_{j=1}^{6} \alpha_{ij}(x_j - p_{ij})^2] \tag{20}$$

with $0 \leq x_i \leq 1$ $(i = 1, \dots, 6)$; $\mathbf{c} = (1.0, 1.2, 3.0, 3.2)^\top$, and α_{ij} and p_{ij} given in Table 2, a global minimum at $(0.20169, 0.150011, 0.476874, 0.275332, 0.311652, 0.6573)^\top$ with $w = -3.32237$, and five additional local minima.

Table 2: Parameters α_{ij} and p_{ij} of the Hartmann-6 function

α_{ij}	10.0	3.0	17.0	3.5	1.7	8.0
	0.05	10.0	17.0	0.1	8.0	14.0
	3.0	3.5	1.7	10.0	17.0	8.0
	17.0	8.0	0.05	10.0	0.1	14.0
p_{ij}	0.1312	0.1696	0.5569	0.0124	0.8283	0.5886
	0.2329	0.4135	0.8307	0.3736	0.1004	0.9991
	0.2348	0.1451	0.3522	0.2883	0.3047	0.6650
	0.4047	0.8828	0.8732	0.5743	0.1091	0.0381

For each of these four test functions we consider $T = 100$ "new" points $\mathbf{x}_t$ $(t = 1, \dots, 100)$, given k old points These new points give the 100 estimated modified CVs $\widehat{\nu}(\mathbf{x}_t)$ and the 100 estimated scale-sensitive variances $\widehat{\sigma}^2(\mathbf{x}_t)$. We experiment with six values for k; namely, k is $10d$ (following Loeppky et al. (2009)'s rule-of-thumb), $20d$, $30d$, $40d$, $50d$, and $100d$. Moreover we experiment with three bootstrap sample sizes B; namely, 100, 5,000, and 20,000. Figure 1 gives boxplots (based on $T = 100$ observations) for the effects of CS versus CK on $\widehat{\nu}(\mathbf{x}_t)$ and $\widehat{\sigma}^2(\mathbf{x}_t)$; these effects are quantified through $\widehat{\nu}_{\text{CS}}(\mathbf{x}_t)$ - $\widehat{\nu}_{\text{CK}}(\mathbf{x}_t)$ and $\widehat{\sigma}^2_{\text{CS}}(\mathbf{x}_t)$ - $\widehat{\sigma}^2_{\text{CK}}(\mathbf{x}_t)$. This figure gives boxplots only for $k = 10d$ and $B = 5000$, but the other B values give similar plots. *These boxplots do not support our conjecture* that $\widehat{\nu}_{\text{CS}}(\mathbf{x}_t)$ - $\widehat{\nu}_{\text{CK}}(\mathbf{x}_t)$ or $\widehat{\sigma}^2_{\text{CS}}(\mathbf{x}_t)$ - $\widehat{\sigma}^2_{\text{CK}}(\mathbf{x}_t)$ increase as the dimensionality d increases (d = 1, 2, 3, 6 in the four test functions in these plots). Surprisingly, the plot for $d = 6$ suggests that—for some of the 100 new points—CS gives a lower variance estimate than CK does; i.e., the boxplots display some values lower than zero (zero is one of the values displayed on the y-axis).

We therefore further investigate the Hartmann-6 case; i.e., we check if $k = 10d$ is so small that it gives a biased estimator. Figure 2, however, suggests that for bigger k (and B = 20,000)—in half of the new points—CS does not give higher $\widehat{\nu}_{\text{CS}}(\mathbf{x}_t)$ - $\widehat{\nu}_{\text{CK}}(\mathbf{x}_t)$ or higher $\sigma^2_{\text{CS}}(\mathbf{x}_t)$ - $\widehat{\sigma}^2_{\text{CK}}(\mathbf{x}_t)$ than CK does; i.e., these boxplots show medians that remain close to zero, for all six k values. Of course, these boxplots show less variation (around zero) as k increases (so the sample error decreases).

3 Confidence intervals for Kriging predictors

In an appendix we summarize some basic statistical theory on CIs. In this section we present experiments with a GP to study CIs for Kriging predictors. The preceding sections suggest that $\widehat{\sigma}^2_{\text{CS}} \approx \widehat{\sigma}^2_{\text{BK}}$, so we do not present results for BK but focus on the following three alternatives:

1. The literature on CK (implicitly) uses the following two-sided symmetric $(1-\alpha)$ CI for the CK predictor at point $\mathbf{x}_0$:

$$\text{CI}_{\text{CK}}\colon\ \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) \pm z_{\alpha/2}\widehat{\sigma}_{\text{CK}} \tag{21}$$

   where $\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$ and $\widehat{\sigma}^2_{\text{CK}}$ were defined in (7) and (8).

2. CS with $\widehat{\sigma}^2_{\text{CK}}$ in (21) replaced by $\widehat{\sigma}^2_{\text{CS}}$ (defined in (14)) and with the CK point predictor $\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$:

$$\text{CI}_{\text{CS}}\colon\ \widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}}) \pm z_{\alpha/2}\widehat{\sigma}_{\text{CS}}. \tag{22}$$

3. CS with a distribution-free two-sided asymmetric CI based on the percentile method applied to $\widehat{Y}_{\text{CS}}(\mathbf{x}_0, b)$ (defined in (13)) so this CI is

$$\text{CI}_{\text{percentile}} : [\widehat{Y}_{\text{CS};(B\alpha/2)}(\mathbf{x}_0), \widehat{Y}_{\text{CS};(B(1-\alpha/2))}(\mathbf{x}_0)] \tag{23}$$

   where the subscript $(\cdot)$ is the usual symbol for order statistics (resulting from sorting the B values from low to high); we select B such that $B\alpha/2$ and $B(1-\alpha/2)$ are integers.

Note: If $\widehat{\sigma}^2_{\text{CK}} < \widehat{\sigma}^2_{\text{CS}}$, then alternative 2 gives a longer CI and hence a higher coverage. In this alternative we use z instead of t_{B-1} because typically B is so big that $t_{B-1} \approx z$; moreover, alternative 1 also uses z; also see the Note at the very end of this section. It makes no sense to apply alternative 3 to BK, because BK gives predictions at the k old points that do not equal the observed old outputs.

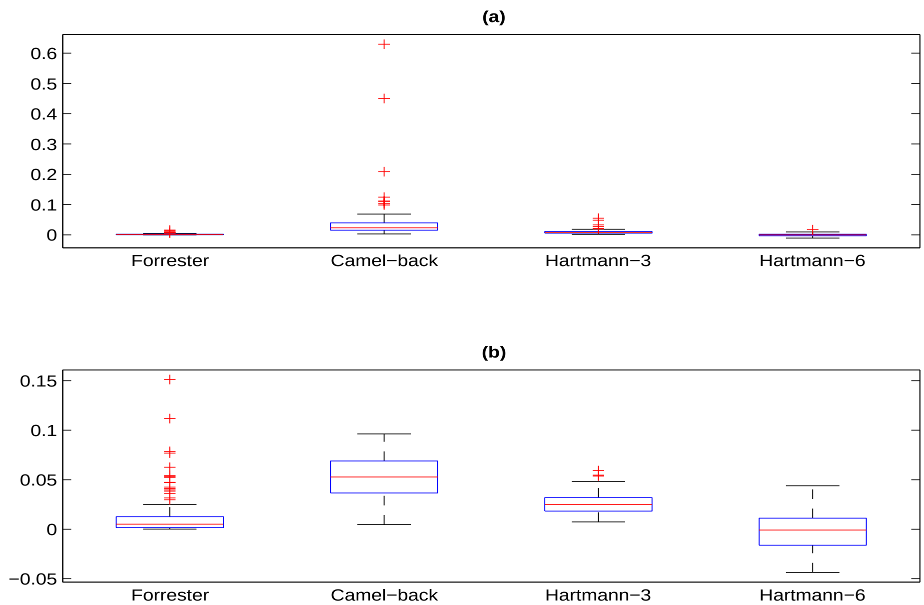


Figure 1: Estimated predictor variance $\widehat{\sigma}^2$ and modified coefficient of variation $\widehat{\nu}$ for CK and CS, in 100 new points t, for four test functions: (a) $\widehat{\nu}_{\mathrm{CS}}(\mathbf{x}_t)-\widehat{\nu}_{\mathrm{CK}}(\mathbf{x}_t)$ and (b) $\widehat{\sigma}^2_{\mathrm{CS}}(\mathbf{x}_t)-\widehat{\sigma}^2_{\mathrm{CK}}(\mathbf{x}_t)$

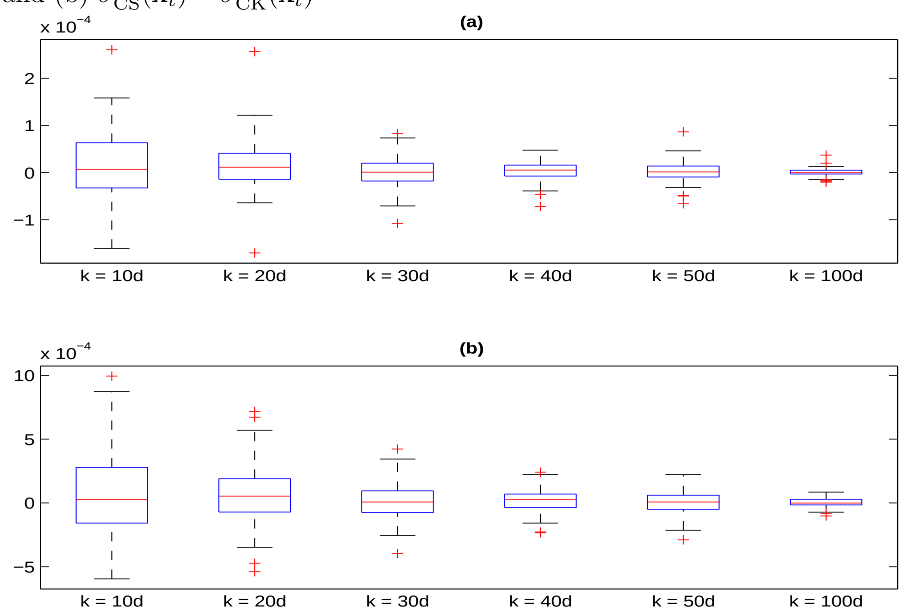


Figure 2: Effects of initial sample size k for Hartmann-6 test function, on the estimated predictor variance $\widehat{\sigma}^2$ and modified coefficient of variation $\widehat{\nu}$ for CK and CS, in 100 new points t: (a) $\widehat{\nu}_{\mathrm{CS}}(\mathbf{x}_t)-\widehat{\nu}_{\mathrm{CK}}(\mathbf{x}_t)$ and (b) $\widehat{\sigma}^2_{\mathrm{CS}}(\mathbf{x}_t)-\widehat{\sigma}^2_{\mathrm{CK}}(\mathbf{x}_t)$

Given a nominal coverage of $1 - \alpha$, we focus on the estimated expected coverage $1 - E(\widehat{\alpha})$ and the estimated expected length $E(l)$ of a CI. In our experiments we try to make the Kriging predictor less biased (in the section on EGO we shall return to the test functions of Section 2.4 for which GPs are only approximations so bias is present, even if we ignore the nonlinearity of the Kriging predictor with plugged-in parameter estimates $\widehat{\boldsymbol{\psi}}$). We select the following example, inspired by Kleijnen et al. (2012). We assume two inputs so $d = 2$. This choice implies that the GP defined in (1) has parameters β_0, τ^2, θ_1, and θ_2, collected in $\boldsymbol{\psi}$. Our choice of $\beta_0 = 127$, $\tau^2 = 11{,}697$, $\theta_1 = 0.11$, and $\theta_1 = 0.16$ is explained as follows.

We select these values after fitting the GP to the camel-back function (18). For this fitting we use 35 input combinations selected through MATLAB's maximin LHS (35 satisfies Loeppky et al. (2009)'s rule-of-thumb for fitting a valid Kriging metamodel, which requires at least 20 points). Obviously, these points define a 35×2 matrix $\mathbf{X}$. Entering this $\mathbf{X}$ into the test function (18) gives the 35-dimensional vector with outputs $\mathbf{w}$. From the resulting I/O set $(\mathbf{X}, \mathbf{w})$ we compute the MLE $\widehat{\boldsymbol{\psi}}$. This $\widehat{\boldsymbol{\psi}}$ serves as the true $\boldsymbol{\psi}$ in our example.

To estimate the coverage and length of a specific alternative CI, we must select k (# old points). We present experimental results for several values of k; namely, 5, 10, 20, and 30. Actually, we select a $(k + 1) \times 2$ matrix $\mathbf{X}$ because we select one new point besides the k old points. We again use MATLAB's maximin LHS to generate $\mathbf{X}$ for various k. From this $\mathbf{X}$ we randomly select the new point avoiding extrapolation because Kriging is known to give a bad extrapolator.

Now we sample the outputs of these k old points plus the new point from the Gaussian distribution (4) with $\Sigma_M = \tau^2 R_M(\theta_1, \theta_2, \mathbf{X})$. The k old points implied by $\mathbf{X}$—together with the corresponding k old outputs $\mathbf{w}$—give $\widehat{\boldsymbol{\psi}}$ (MLE). This $\widehat{\boldsymbol{\psi}}$ gives the predicted new output $\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$ and $\widehat{\sigma}^2_{\mathrm{CK}}$; CK uses this $\widehat{Y}(\mathbf{x}_0, \widehat{\boldsymbol{\psi}})$ and $\widehat{\sigma}^2_{\mathrm{CK}}$ to compute the CI in (21). This $\widehat{\boldsymbol{\psi}}$ is also used by CS to obtain the parametric CI (22) and the distribution-free CI (23), using B bootstrap samples.

Obviously, a CI either covers the true output $y(\mathbf{x}_0)$—sampled from (4)—or misses it we observe the *Bernoulli* variable (say) $c_r \in \{0,\ 1\}$ where $r = 1, \ldots, M$ with M denoting the number of *macroreplications* (macroreplications use different non-overlapping streams of pseudorandom numbers, while fixing all other experimental factors such as B and $\mathbf{x}_0$; obviously macroreplications give outputs that are IID). So, $P(c_r = 0) = E(\widehat{\alpha}) = p$ with $\widehat{\alpha} = \sum c_r/M$ so $\mathrm{Var}(\widehat{\alpha}) = p(1 - p)/M$, because $\sum c_r$ is a binomial variable. The CI's mean length is estimated through $\bar{l} = \sum l_r/M$ so $\mathrm{SE}(\bar{l}) = \sqrt{\widehat{\sigma}^2(l)/M}$ with $\widehat{\sigma}^2(l) = \sum(l_r - \bar{l})^2/(M - 1)$.

We obtain results when α is 0.10 and B is 100 and 2,000 respectively. Because the results for these two B values are similar, Table 3 displays results for $B = 100$ only. We prefer the CI with the shortest length, unless this CI gives too low coverage. In this table, CK with $\widehat{\sigma}_{\mathrm{CK}}$ gives shorter lengths than CS with $\widehat{\sigma}_{\mathrm{CS}}$, and yet CK gives estimated coverages that are not significantly lower; this lack of significance is determined by the SEs displayed below the estimated $\widehat{\alpha}$ and $\bar{l}$. It is well known that variance estimators—such as SEs—show more variability than mean estimators—such as $\widehat{\alpha}$ and $\bar{l}$—do. CS with the percentile method gives longer lengths than CK, but its coverage is not significantly better than CK. We observe that for each of the three alternative CIs, the length tends to decrease as k increases (so the new point has neighbors that are closer, which have outputs that are more correlated with the output of the new point). Altogether the results in this table do not convince us that CS is superior, so we recommend CK when predicting a new output. Such a prediction is made in sensitivity analysis (as opposed to simulation-optimization; see the next section).

Table 3: Coverage and length of 90 % CI, for k old points and three alternative CIs defined in (21) using $\hat{\sigma}_{\mathrm{CK}}$, (22) using $\hat{\sigma}_{\mathrm{CS}}$, and (23) using percentiles (SE in parentheses)

k	Coverage for alternative			Length for alternative		
	$\hat{\sigma}_{\mathrm{CK}}$	$\hat{\sigma}_{\mathrm{CS}}$	percentiles	$\hat{\sigma}_{\mathrm{CK}}$	$\hat{\sigma}_{\mathrm{CS}}$	percentiles
5	0.68	0.77	0.72	46.12	60.33	60.81
	(0.05)	(0.04)	(0.05)	(3.61)	(3.48)	(3.48)
10	0.86	0.98	0.94	1.33	5.55	2.32
	(0.03)	(0.01)	(0.02)	(0.44)	(0.62)	(0.55)
20	0.82	0.88	0.85	5.1E-04	2.5E-03	5.8E-04
	(0.04)	(0.03)	(0.04)	(1.9E-05)	(8.5E-04)	(2.2E-05)
30	0.88	0.94	0.87	8.0E-04	1.1E-03	8.5E-04
	(0.03)	(0.02)	(0.03)	(3.0E-05)	(5.5E-05)	(3.5E-05)

Note: We also experiment with a Studentized version of CK's CI; namely, in (21) we replace $z_{\alpha/2}$ by $t_{k-(2+d)\alpha/2}$ where $2+d$ is the number of estimated GP parameters. This version is only a heuristic because we do not know the correct value for the degrees of freedom of t. Our experimental results show that for $k=5$ the CI is extremely long compared with (21); for higher k the results are similar to the results for (21).

4 Efficient global optimization

First we present several variants of EGO using CK, BK, or CS. Next we present experiments with these variants.

4.1 EGO variants using CK, BK, or CS

Suppose the goal of the simulation optimization is to minimize the simulation output w. EGO with CK consist of the following five steps.

1. Fit a Kriging metamodel $Y(\mathbf{x})$ to the old I/O data $(\mathbf{X}, \mathbf{w})$.

2. Find the minimum output observed (simulated) so far: $f_{\min} = \min_{1 \le i \le k} w(\mathbf{x}_i)$.

3. Find $\widehat{\mathbf{x}}_{\text{opt}}$, which denotes the estimate of $\mathbf{x}_0$ that *maximizes* the so-called *expected improvement (EI)*:
$$\text{EI}(\mathbf{x}) = E\left[\max(f_{\min} - Y(\mathbf{x}), 0)\right]. \tag{24}$$
Assuming $Y(\mathbf{x}) \sim \text{N}\left(\widehat{Y}(\mathbf{x}), \widehat{\sigma}^2_{\text{CK}}(\mathbf{x})\right)$ gives the closed-form expression (derived in Jones et al. (1998)) for the estimator of (24):
$$\widehat{\text{EI}}(\mathbf{x}) = \left(f_{\min} - \widehat{Y}(\mathbf{x})\right) \Phi\left(\frac{f_{\min} - \widehat{Y}(\mathbf{x})}{\widehat{\sigma}_{\text{CK}}(\mathbf{x})}\right) + \widehat{\sigma}_{\text{CK}}(\mathbf{x}) \phi\left(\frac{f_{\min} - \widehat{Y}(\mathbf{x})}{\widehat{\sigma}_{\text{CK}}(\mathbf{x})}\right) \tag{25}$$
with $\widehat{Y}(\mathbf{x})$ defined in (7) and $\widehat{\sigma}_{\text{CK}}(\mathbf{x})$ being the square root of $\widehat{\sigma}^2_{\text{CK}}(\mathbf{x})$ defined in (8); Φ and ϕ are the usual symbols for the cumulative distribution function and probability density function of the standard normal variable z.

4. Run the simulation model with $\widehat{\mathbf{x}}_{\text{opt}}$ found in step 3, to find $w(\widehat{\mathbf{x}}_{\text{opt}})$.

5. Fit a new Kriging metamodel to the old I/O data of step 1 and the new I/O of step 4. Update k and return to step 2 if the stopping criterion is not yet satisfied. A stopping criterion will be discussed in Section 4.2.

To find $\widehat{\mathbf{x}}_{\text{opt}}$ in step 3, EGO can choose among many optimizers. For example, Jones et al. (1998) use a branch-and-bound algorithm, whereas Viana et al. (2013) use an evolutionary algorithm. Other authors use a set of candidate points (e.g., selected through MATLAB's maxmin LHS), and use the candidate point that maximizes $\widehat{\text{EI}}$ as $\widehat{\mathbf{x}}_{\text{opt}}$; see Kleijnen et al. (2012); Scott et al. (2010); Taddy et al. (2009); Echard et al. (2011). In the next section we shall present results for a set of candidate points only (because we got into numerical problems when applying (Forrester et al., 2008, p. 83)'s genetic algorithm (GA)—which is a global optimizer—followed by MATLAB's `fmincon`—which is a local optimizer).

Note: The predictor $\widehat{Y}(\mathbf{x}_0)$ following from (7) depends on the new point $\mathbf{x}_0$ only through $\widehat{\mathbf{\Sigma}}_M(\mathbf{x}_0, \cdot)$; i.e., all k old points $\mathbf{x}$ use the same $\widehat{\beta}_0$ and $\widehat{\mathbf{\Sigma}}_M^{-1}[Y(\mathbf{x}_i) - \widehat{\beta}_0 \mathbf{1}_k]$ $(i = 1, \ldots, k)$.

Besides EGO with CK, Kleijnen et al. (2012) presents EGO with BK, replacing $\widehat{\sigma}_{\text{CK}}$ in (25) by $\widehat{\sigma}_{\text{BK}}$ following from (10). We now also present two CS variants. In one CS variant we replace $\widehat{\sigma}_{\text{CK}}$ by $\widehat{\sigma}_{\text{CS}}$ following from (14) and replace $\widehat{Y}(\mathbf{x})$ by the median $\widehat{Y}_{\text{CS};(B/2)}(\mathbf{x})$, which follows from (13); we select B such that $B/2$ is integer (Kleijnen and Mehdad (2013) do not use this median). It is well-known that in general the median is a robust estimator in case of nonnormality; see Andrews et al. (1972). In the other CS variant we introduce a *distribution-free* estimator of EI:
$$\widehat{\text{EI}}_{\text{CS}}(\mathbf{x}) = \frac{\sum_{b=1}^{B} \max(f_{\min} - \widehat{Y}_{\text{CS}}(\mathbf{x}, b), 0)}{B} \tag{26}$$

where $\widehat{Y}_{\text{CS}}(\mathbf{x}, b)$ follows from (13). We point out that (26) gives $\widehat{\text{EI}}_{\text{CS}}(\mathbf{x}) = 0$ if the smallest of the B predicted values for $\mathbf{x}$ exceeds $f_{\min}$; had we assumed normally distributed $\widehat{Y}_{\text{CS}}(\mathbf{x})$ (like in (24)), then $\widehat{\text{EI}}_{\text{CS}}(\mathbf{x})$ would have been a small positive value (because the normal distribution has support from $-\infty$ to ∞). Our experiments (detailed in the next subsection) suggest that this variant does not improve CK.

4.2 Experiments with EGO variants

We use numerical experiments to evaluate the EGO variants described in Section 4.1. These experiments use the same four test functions as Kleijnen et al. (2012) used, which we also used in Section 2.4; the Hartmann-3 and Hartmann-6 functions are also used in recent articles such as Viana et al. (2013).

To measure the performance of a variant, we use k which denotes the number of simulated input combinations that the variant needs to estimate the optimal input combination. As the stopping criterion we select $\widehat{\text{EI}} < 10^{-20}$ or k reaching a limit; following Kleijnen et al. (2012) we select this limit to be 11 for Forrester, 61 for camel-back, 65 for Hartmann-3, and 101 for Hartmann-6. We select this stopping criterion to avoid stopping "early"; i.e., with this criterion we can observe possible convergence of the variant's search.

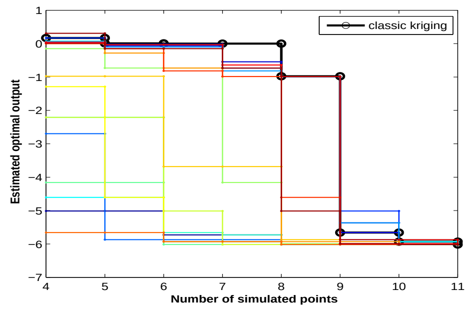
To implement the variants, we use MATLAB's DACE. To compute the MLE $\widehat{\boldsymbol{\theta}}$, we first apply Forrester et al. (2008)'s GA and then we use the resulting GA values to initialize DACE's Hooke-Jeeves's algorithm, for the first three test functions; because of numerical complications for the Hartmann-6 function, we do not apply this GA but we use DACE with its default values.

For BK and CS we select $B = 100$. Because bootstrapping implies sampling, we obtain macroreplications; we decide to use $M = 10$ macroreplications, except for Forrester's function with $d = 1$ for which we use $M = 20$. For the Hartmann-6 function we follow Jones et al. (1998), transforming the output w to $-\ln(-w)$.

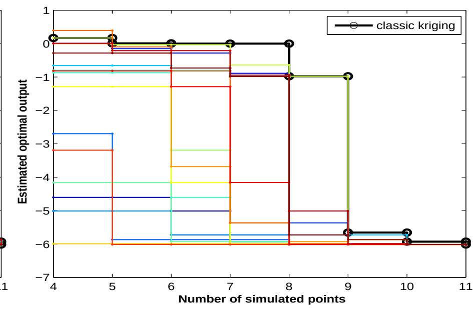
We select the set of candidate points, following Kleijnen et al. (2012); i.e., for Forrester's function we use a grid with distance 0.01 between consecutive input locations so we get 98 candidate points; for the camel-back function we select 200 candidate points through the maximin LHS design found on `http://www.spacefillingdesigns.nl/`; for Hartmann-3 we use MATLAB's maximin LHS design with 300 points; for Hartmann-6 we use MATLAB's maximin LHS design with 500 points.

Note: Section 2 suggests that in many applications we have $\widehat{\sigma}^2_{\text{CK}} < \widehat{\sigma}^2_{\text{BK}} \approx \widehat{\sigma}^2_{\text{CS}}$. EGO, however, is not using the magnitude of the predictor variance $\widehat{\sigma}^2[\widehat{Y}(\mathbf{x})]$, but searches for the point $\mathbf{x}$ that maximizes $\widehat{\sigma}^2[\widehat{Y}(\mathbf{x})]$ if the predicted values for alternative $\mathbf{x}$ were the same. We find that indeed the EGO variants may select different $\widehat{\mathbf{x}}_{\text{opt}}$.

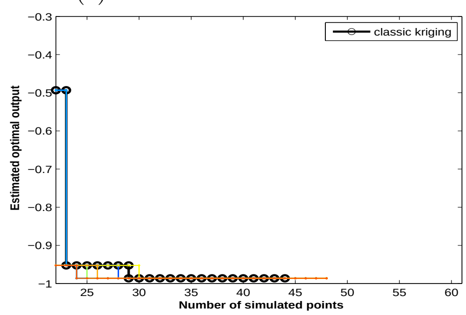
We observe that Kleijnen and Mehdad (2013) has already presented detailed results for Forrester's function. Figure 3 displays $f_{\min}(k) = \min w(\mathbf{x}_i)$ $(1 \leq i \leq k)$, which denotes the estimated optimal simulation output after k simulated input combinations; horizontal lines mean that the most recent simulated point $\widehat{\mathbf{x}}_{\text{opt}}(k)$ does not give a lower output than a preceding point. This



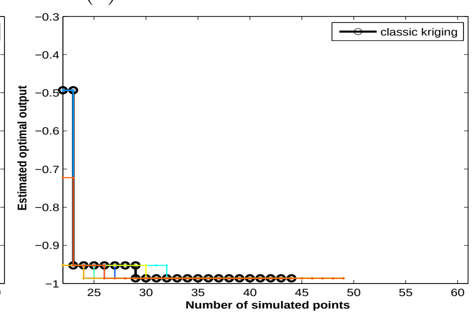
(a) BK in Forrester's function



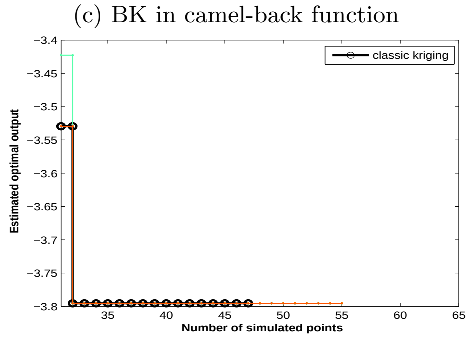
(b) CS in Forrester's function



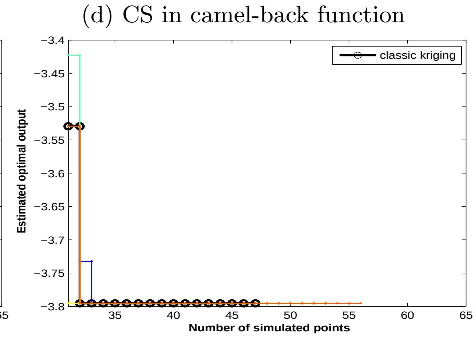
(c) BK in camel-back function



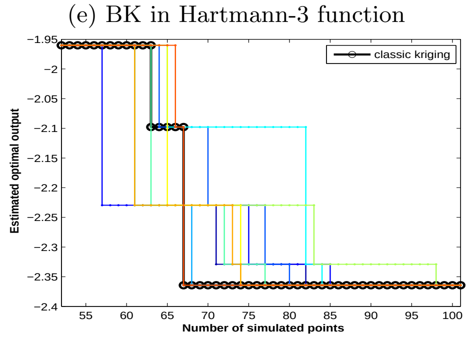
(d) CS in camel-back function



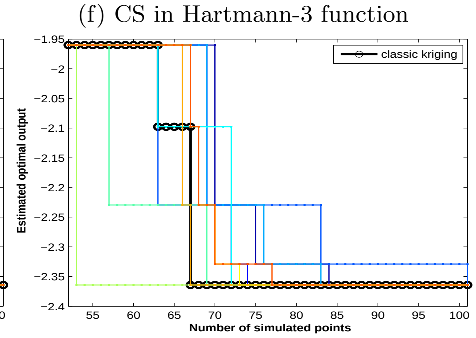
(e) BK in Hartmann-3 function



(f) CS in Hartmann-3 function



(g) BK in Hartmann-6 function



(h) CS in Hartmann-6 function

Figure 3: Estimated optimal output (y-axis) after k simulated input combinations (x-axis) for four test functions (top to bottom panels) for BK (left panels) and CS (right panels) versus CK (legend within each panel)

figure shows $f_{\min}(k)$ for BK and CS relative to CK. More specifically, the black step function with circles represents $f_{\min}(k)$ for CK; colors are used only in the online version. The colored step functions represent $f_{\min}(k)$ for BK (left-hand panels) or CS (right-hand panels). Actually, a colored step function may represent more than one macroreplication; e.g., for Forrester's function we obtain $M = 20$ macroreplications, but in the upper two panels we cannot distinguish 20 colored step functions.

We conclude that for *Forrester's* function all three EGO variants give the same estimated optimal I/O for k =11; namely, $\widehat{w}_{\text{opt}} = w(\widehat{x}_{\text{opt}})$ = -6.017 (the figure does not show that $\widehat{x}_{\text{opt}} = 0.76$; the true values for continuous x are $x_{\text{opt}} = 0.7572$ and $w_{\text{opt}} = -6.02074$). For expensive simulations with small sample sizes, this asymptotic solution may not be relevant. The detailed data behind the figure reveal that CK performs better than CS only in one of the twenty macroreplications, when $k = 4$. CK performs better than BK in one macroreplication when $k = 4$, three macroreplications when $k = 9$, and two macroreplications when $k = 10$.

For the *camel-back* function we start with $k = 21$ points. For $22 \leq k \leq 28$ BK and CS perform better than CK, in more than half of the 10 macroreplications. For the *Hartmann-3* function we start with 30 points. CK seems quite robust. For the *Hartmann-6* function we start with only 51 points, as Jones et al. (1998) does. For $61 \leq k \leq 66$ BK performs better than CK, in more than half of the 10 macroreplications. CS does not perform better in more than half the macroreplications.

In practice we do not know whether the simulation model has an I/O function that resembles one of the four test functions in this figure; therefore practitioners may wish to stick to CK.

5 Conclusions and future research

Classic Kriging (CK) estimates the variance of its predictor by plugging-in the estimated GP parameters $\widehat{\boldsymbol{\psi}}$; the problem is that this variance is biased. As solutions we study bootstrapped Kriging (BK) and conditional simulation (CS). We prefer CS over BK because CS is computationally and conceptually simpler, and CS gives better predictions near old points. A confidence interval (CI) may be either parametric using the estimated variance of the Kriging predictor or distribution-free using the bootstrap's percentile method. Experimentally we find that BK and CS give predicted variances that do not differ significantly from each other, but that may be significantly bigger than the classic estimate. Nevertheless, BK and CS do not give CIs that are significantly better than CK. We also use these alternative predictor variances in EGO. Our experiments with several test functions suggest that EGO with BK or CS may or may not perform better than CK; therefore practitioners may prefer CK.

In future research, we shall adapt EGO for random simulation with replications, using distribution-free bootstrapping (instead of parametric bootstrapping assuming a Gaussian distribution for the Kriging metamodel). We shall also consider multiple simulation outputs leading to constrained optimization.

Acknowledgments

We thank Inneke van Nieuwenhuyse (K.U. Leuven, Leuven, Belgium) for sharing her MATLAB code for EGO using BK and her help with the implementation of that code, Dick den Hertog (Tilburg University) for suggesting the investigation of dimensionality effects on the variance, and three anonymous reviewers for their very useful comments on a previous version.

References

Andrews, D. F., Bickel, P. J., Hampel, F. R., Huber, P., Rogers, W., and Tukey, J. (1972). *Robust Estimates of Location: Survey and Advances*. Princeton University Press.

Ankenman, B. E., Nelson, B. L., and Staum, J. (2010). Stochastic Kriging for simulation metamodeling. *Operations Research*, 58:371–382.

Den Hertog, D., Kleijnen, J., and Siem, A. (2006). The correct Kriging variance estimated by bootstrapping. *Journal of The Operational Research Society*, 57:400–409.

Echard, B., Gayton, N., and Lemaire, M. (2011). Ak-mcs: An active learning reliability method combining Kriging and Monte Carlo Simulation. *Structural Safety*, 33(2):145–154.

Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability. Chapman & Hall.

Forrester, A., Sóbester, A., and Keane, A. (2008). *Engineering Design via Surrogate Modelling: A Practical Guide*. Wiley.

Goel, T., Haftka, R., Queipo, N., and Shyy, W. (2006). Performance Estimate and Simultaneous Application of Multiple Surrogates. In *11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Multidisciplinary Analysis Optimization Conferences. American Institute of Aeronautics and Astronautics.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492.

Kleijnen, J. P. C. (2008). *Design and Analysis of Simulation Experiments*. Springer-Verlag.

Kleijnen, J. P. C. (2009). Kriging metamodeling in simulation: A review. *European Journal of Operational Research*, 192(3):707–716.

Kleijnen, J. P. C. and Mehdad, E. (2013). Conditional simulation for efficient global optimization. In *Proceedings of the 2013 Winter Simulation Conference (WSC)*, pages 969–979.

Kleijnen, J. P. C., Van Beers, W., and Van Nieuwenhuyse, I. (2012). Expected improvement in efficient global optimization through bootstrapped Kriging. *Journal of Global Optimization*, 54:59–73.

Loeppky, J. L., Sacks, J., and Welch, W. J. (2009). Choosing the sample size of a computer experiment: A practical guide. *Technometrics*, 51:366–376.

Lophaven, S. N., Nielsen, H. B., and Sondergaard, J. (2002). *DACE: a MATLAB Kriging toolbox, version 2.0*. IMM Technical University of Denmark, Lyngby, Denmark.

Picheny, V., Ginsbourger, D., Richet, Y., and Caplin, G. (2013). Quantile-based optimization of noisy computer experiments with tunable precision (including comments). *Technometrics*, 55(1):2–36.

Scott, W. R., Powell, W. B., and Simao, H. P. (2010). Calibrating simulation models using the knowledge gradient with continuous parameters. In *Proceedings of the 2010 Winter Simulation Conference (WSC)*, pages 1099–1109.

Taddy, M. A., Lee, H. K. H., Gray, G. A., and Griffin, J. D. (2009). Bayesian guided pattern search for robust local optimization. *Technometrics*, 51(4):389–401.

Törn, A. and Žilinskas, A. (2008). *Global Optimization (Lecture Notes in Computer Science)*. Springer, 1989 edition.

Viana, F. A. C., Haftka, R. T., and Watson, L. T. (2013). Efficient global optimization algorithm assisted by multiple surrogate techniques. *Journal of Global Optimization*, 56(2):669–689.

Wackernagel, H. (2003). *Multivariate Geostatistics: An Introduction with Applications*. Springer-Verlag, New York.

Xie, W., Nelson, B., and Staum, J. (2010). The influence of correlation functions on stochastic kriging metamodels. In *Proceedings of the 2010 Winter Simulation Conference (WSC)*, pages 1067–1078.

Yin, J., Ng, S. H., and Ng, K. M. (2010). A bayesian metamodeling approach for stochastic simulations. In *Proceedings of the 2010 Winter Simulation Conference (WSC)*, pages 1055–1066.

Appendix: CI basics

We start with the classic CI for the *mean* (say) $E(Y) = \mu_y$ of $Y \sim \text{NIID}(\mu_y, \sigma_y^2)$—where NIID stands for normally, identically, and independently distributed (IID)—given k observations y_i $(i = 1, \ldots, k)$:

$$\text{CI}_{E(Y)}\colon\ \overline{Y} \pm t_{k-1;\alpha/2}\widehat{\sigma}_{\overline{y}} \tag{27}$$

with the sample mean $\overline{Y} = \sum_{i=1}^{k} Y_i/k$ and sample variance $\widehat{\sigma}_{\overline{y}}^2 = \widehat{\sigma}_y^2/k$ where $\widehat{\sigma}_y^2 = \sum_{i=1}^{k}(Y_i - \overline{Y})^2/(k-1)$. This sample mean and sample variance are unbiased estimators, whereas the MLE (Kriging uses MLE) of σ_y^2 would be $\widehat{\sigma}_y^2(k-1)/k$ so the MLE underestimates the true variance and MLE gives a CI with coverage lower than $1-\alpha$. There is much research on the robustness of the CI in (27). For example, Andrews et al. (1972) conclude that this CI is optimal if and only if all its assumptions hold; i.e., in case of nonnormality this CI is not optimal, and an alternative point estimator is the sample median $Y_{(k/2)}$. Obviously, the sample median is less sensitive to outliers. A simple *distribution-free* $1-\alpha$ CI is $[Y_{(0.05k)}, Y_{(0.95k)}]$, which follows from the percentile method in Efron and Tibshirani (1993); this CI assumes that $Y \sim \text{IID}(\mu_y, \sigma_y^2)$, and $0.05k$ and $0.95k$ are integers. Note that (27) is the basis of the classic CI defined in (21).

Next we consider a CI for the *linear regression* predictor. The linear regression model is

$$Y(\mathbf{x}, \boldsymbol{\beta}) = \mathbf{f}(\mathbf{x})^\top \boldsymbol{\beta} + \epsilon(\mathbf{x}) \tag{28}$$

where $\boldsymbol{\beta}$ is a q-dimensional vector of unknown regression parameters, $\mathbf{f}(\mathbf{x})$ is a q-dimensional vector of known functions of $\mathbf{x}$, and the regression residual $\epsilon(\mathbf{x})$ has zero mean and a variance that may vary with $\mathbf{x}$ and correlations $\text{Corr}[\epsilon(\mathbf{x}),\epsilon(\mathbf{x}')]$ that are positive in case of simulation with common random numbers (CRN); this model is also given by Ankenman et al. (2010). The best linear unbiased estimator (BLUE) of $\boldsymbol{\beta}$ is the generalized least squares (GLS) estimator

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}^\top \mathbf{V}^{-1} \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{V}^{\ -1} \mathbf{Y} \tag{29}$$

where $\mathbf{X}$ is the $k \times q$ input matrix, $\mathbf{V}$ is the positive-definite symmetric $k \times k$ covariance matrix of $Y(\mathbf{x}, \boldsymbol{\beta})$, and $\mathbf{y}$ is the k-dimensional vector of output observations. Obviously (28) and (29) gives the predictor $\widehat{Y}(\mathbf{x}_0,\widehat{\boldsymbol{\beta}}) = \mathbf{f}(\mathbf{x}_0)\widehat{\boldsymbol{\beta}}$. Clearly, the variance of this predictor is

$$\text{Var}[\widehat{Y}(\mathbf{x}_0,\widehat{\boldsymbol{\beta}})] = \mathbf{f}(\mathbf{x}_0)^\top \mathbf{cov}(\widehat{\boldsymbol{\beta}}) \mathbf{f}(\mathbf{x}_0)$$

where (29) implies

$$\mathbf{cov}(\widehat{\boldsymbol{\beta}})\ = (\mathbf{X}^\top \mathbf{V}^{-1} \mathbf{X})^{-1}. \tag{30}$$

The CI for the linear regression (LR) predictor is

$$\text{CI}_{\text{LR}}\colon\ \widehat{Y}(\mathbf{x}_0,\widehat{\boldsymbol{\beta}}) \pm z_{\alpha/2}\sqrt{\text{Var}[\widehat{Y}(\mathbf{x}_0,\widehat{\boldsymbol{\beta}})]}. \tag{31}$$

This CI, however, assumes a known $\mathbf{V}$—besides multivariate normality. In practice, $\mathbf{V}$ is unknown so it is estimated by $\widehat{\mathbf{V}}$, which changes the GLS estimator (29) into a nonlinear estimator.

In practice, analysts often assume $\mathbf{V} = \sigma_y^2\mathbf{I}$, so the GLS estimator in (29) reduces to the ordinary least squares (OLS) estimator $\widehat{\boldsymbol{\beta}}_{\text{OLS}} = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{Y}$ and $\mathbf{cov}(\widehat{\boldsymbol{\beta}}_{\text{OLS}}) = \sigma_y^2(\mathbf{X}^\top\mathbf{X})^{-1}$ where σ_y^2 is estimated through the mean squared residuals (MSR), $(\widehat{\mathbf{Y}} - \mathbf{Y})^\top(\widehat{\mathbf{Y}} - \mathbf{Y})/(k-q)$ assuming $k > q$. Finally, (31) is replaced by

$$\text{CI}_{\text{OLS}}\colon\ \widehat{Y}(\mathbf{x}_0,\widehat{\boldsymbol{\beta}}_{\text{OLS}}) \pm t_{n-q;\alpha/2}\sqrt{\widehat{\text{Var}}[\widehat{Y}(\mathbf{x}_0,\widehat{\boldsymbol{\beta}}_{\text{OLS}})]}, \tag{32}$$

which resembles (27).

We point out that linear regression uses the unbiased least squares (LS) estimator, whereas Kriging uses MLE. Bootstrapping in linear regression is also discussed by (Efron and Tibshirani, 1993, pp. 70-80).